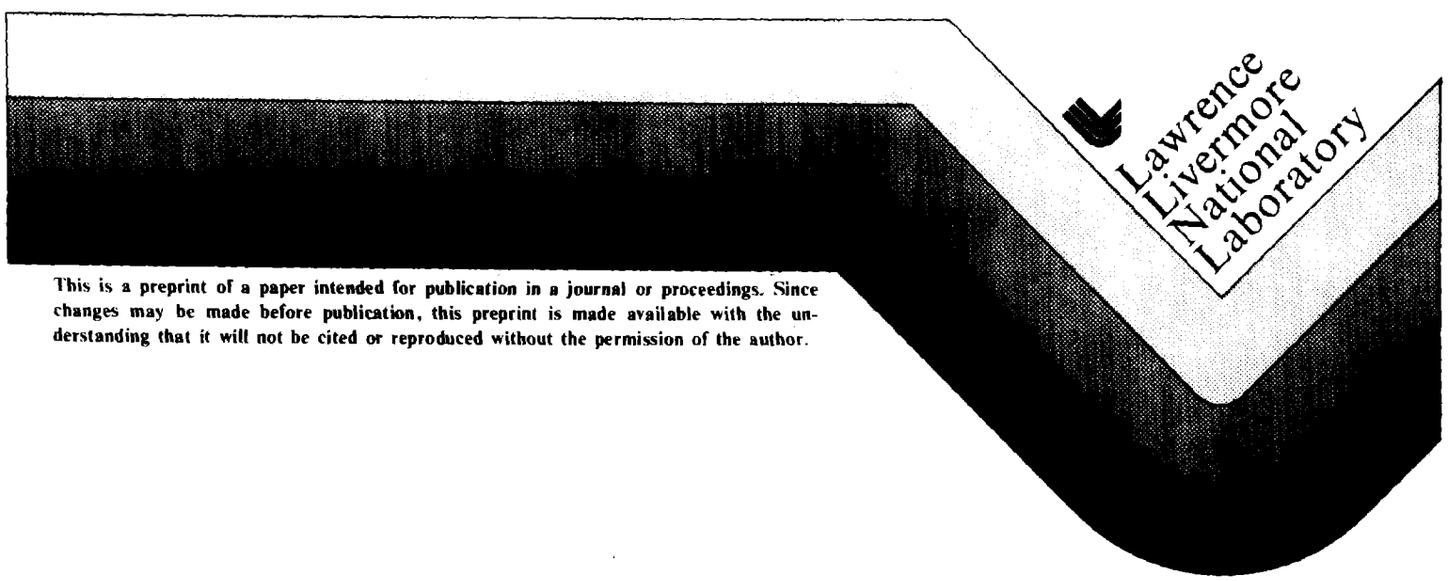
UCRL- 93589
PREPRINT

IMPLEMENTATION OF THE GMRES AND ARNOLDI METHODS USING HOUSEHOLDER TRANSFORMATIONS

Homer F. Walker

This paper was prepared for submittal to
SIAM Journal on Scientific and Statistical
Computing

October 1985



Implementation of the GMRES and Arnoldi Methods Using Householder Transformations

by

Homer F. Walker

Department of Mathematics

University of Houston-University Park

Houston, Texas 77004

Abstract. The standard implementations of the GMRES and Arnoldi methods for solving large linear systems involve Gram-Schmidt processes which are potential sources of significant numerical error. Alternative implementations are outlined here in which orthogonalization by Householder transformations replaces the Gram-Schmidt processes. These implementations require essentially the same storage and arithmetic as the standard implementations, and they should have better numerical properties.

Abbreviated title: Implementations Using Householder Transformations

Key words: GMRES method, Arnoldi method, incomplete orthogonalization method, Householder transformations, Gram-Schmidt process

Implementations of the GMRES and Arnoldi Methods Using Householder Transformations *

by

Homer F. Walker

Department of Mathematics

University of Houston-University Park

Houston, Texas 77004

1. Introduction.

In recent years, there has been a great deal of interest in iterative methods for solving large linear systems of equations

$$Ax = b \tag{1.1}$$

in which $A \in R^{nxn}$ is non-symmetric. Of interest here are two of these methods, the generalized minimal residual (GMRES) method of Saad and Schultz [6] and the Arnoldi method described by Saad [4]. The reader is referred to [4] and [6] for a full discussion of these methods and their standard implementations. For examples of the successful application of these methods to problems arising from the numerical solution of ordinary and partial differential equations, see Brown and Hindmarsh [2] and Wigton, Yu and Young [7].

The standard implementations of the GMRES and Arnoldi methods center around the Arnoldi orthogonalization algorithm for inductively generating orthonormal bases $\{ v_1, \ldots, v_m \}$ of Krylov Subspaces

---

* This work was supported in part by the U.S. Department of Energy under grant DE-AS05-76ER05046. It was also performed in part under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under Contract No. W-7405-Eng-48, and supported by the DOE Office of Energy Research, Applied Mathematical Sciences Research Program.

K_m = span { r_0, Ar_0, ... , $A^{m-1}r_0$ }, m= 1, 2, ... , where $r_0 = b - Ax_0$ for an initial approximate solution x_0 of (1.1). This algorithm is usually given as follows ($\| \cdot \|$ and $(\cdot,\cdot)$ denote the Euclidean norm and inner product):

Arnoldi orthogonalization:

1. Set $v_1 = r_0 / \| r_0 \|$.
2. For j = 1, 2, ... , do:

$$h_{ij} = (Av_j , v_i), \quad i = 1, 2, \ldots , j,$$

$$\hat{v}_{j+1} = Av_j - \sum_{i=1}^{j} h_{ij} v_i ,$$

$$h_{j+1,j} = \| \hat{v}_{j+1} \| ,$$

$$v_{j+1} = \hat{v}_{j+1} / h_{j+1,j} .$$

This is a Gram-Schmidt process and is numerically untrustworthy. Because of roundoff, there may be severe loss of orthogonality among the computed v_i's. In practice, it is usual to implement Arnoldi orthogonolization using the modified Gram-Schmidt process (see Golub and Van Loan [3]). Mathematically, this is just a rearrangement of the standard process; computationally, it has superior properties.

Unfortunately, even the modified Gram-Schmidt process can fail to perform well if the vectors on which it operates are not sufficiently independent. Indeed, if $S = [s_1, \ldots , s_m]$ is an nxm matrix the columns of which are to be orthonormalized and if $Q = [q_1, \ldots , q_m]$ is the computed result of applying modified Gram-Schmidt to the columns of S using floating point arithmetic with unit rounding error u, then

Bjorck [1] has shown that

$$Q^TQ = I + E, \qquad \| E \| \cong \underline{u}\, \kappa_2(S), \tag{1.2}$$

where the condition number $\kappa_2(S)$ is the ratio of the largest singular value of S to the smallest. Saad [5, p. 214] has suggested that Gram-Schmidt orthogonalization is a major source of errors in practice for the methods of interest here.

There is an alternative orthogonalization procedure based on the use of Householder transformations which is reliable even if the vectors to be orthonormalized are not very independent. A Householder transformation is of the form $P = I - 2vv^T$, where $\| v \| = 1$. Note that $P = P^T = P^{-1}$. Also, note that the action of P on a vector or a matrix can be easily determined using v; in particular, one need not explicitly form or store P in the applications of interest here. For more on the properties and uses of Householder transformations, see Golub and Van Loan [3] . To orthonormalize the columns of $S = [\, s_1, \ldots, s_m \,]$, one can determine Householder transformations $P_1, \ldots, P_m$ such that $P_m \ldots P_1 S = R$, an upper triangular matrix. Then $S = P_1 \ldots P_m R$, and so the matrix Q consisting of the first m columns of $P_1 \ldots P_m$ gives the desired orthonormalization of the columns of S. If Q is computed in floating point arithmetic with unit rounding error $\underline{u}$, then (Bjorck [1])

$$Q^TQ = I + E, \qquad \| E \| \cong \underline{u}\,. \tag{1.3}$$

In view of the greater reliability of orthogonalization based on Householder transformations, it seems worthwhile to consider implementations of the GMRES and Arnoldi methods in which this orthogonalization

replaces the Gram-Schmidt processes in the standard implementations. We outline such implementations here. Storage and arithmetic for these implementations are approximately the same as for the standard implementations. A major difference is that in our implementations, most resources go into the creation and storage of vectors associated with the necessary Householder transformations; it is not necessary to deal explicitly with the orthonormal bases of Krylov subspaces which these transformations implicitly determine. In the standard implementations, most resources go toward explicitly creating and storing these orthonormal bases. In both the standard implementations and in our implementations, there appear upper Hessenberg matrices which must be triangularized. To be consistent with the overall goal of numerical stability, we use Givens rotations for this task in both implementations here. Optionally, one can implement the Arnoldi method using Gaussian elimination with partial pivoting at slightly less expense.

The implementations given here are of the "restarted" or "iterative" versions of the GMRES and Arnoldi methods (see Saad [4], Saad and Schultz [6]). In these versions, the iteration proceeds for no more than a fixed, preset maximum number of times. If an adequate approximate solution has not been found by then, then the method is restarted with the last approximate solution used as the new initial guess. As in the standard implementations, our implementations allow the size of the residual to be determined at each iteration without having to compute the actual approximate solution at each iteration. Thus the size of the residual can be tested at each iteration to determine whether an adequate approximate solution has been reached.

Once an adequate approximate solution has been reached, or the maximum number of iterations has been attained, the approximate solution is explicitly computed. One can, of course, obtain implementations of the "un-restarted" versions of these methods simply by removing the upper bound on the number of iterations in the algorithms given here. It is not so clear how to implement effective "incomplete" Householder-transformation based methods which are analogues of the incomplete orthogonalization method (IOM) of Saad [4]. Several possibilities readily present themselves, but it would obscure the basic issues of interest to speculate on them here.

Notational conventions are as follows: Capital letters denote matrices; lower case letters denote vectors and scalars. Vector components and matrix entries will be indicated by superscripts in parentheses, e.g., $v^{(i)}$ denotes the $i^{\underline{th}}$ component of the vector v. With or without subscripts or other distinguishing marks, the letters H, J, P, Q, and R always indicate matrices of the following respective types: upper Hessenberg, Givens rotation, Householder transformation, orthogonal, and upper triangular. (See Golub and Van Loan [3] for definitions and properties.) The $i^{\underline{th}}$ standard basis vector, i.e., the i^{th}column of the identity matrix I, is denoted by e_i. Dimensions of vectors and matrices and, when appropriate, their (possibly) non-zero elements are implicit from the contexts in which they appear. For example, if R is pxq upper triangular and we write $R = [\ ce_1,\ H,\ h\ ]$, then H must be px(q-2) upper Hessenberg and h must be a p-vector with zero components after the $q^{\underline{th}}$. For $k = 1, 2, \ldots$, we denote $K_k = [\ r_0,\ Ar_0,\ \ldots\ ,\ A^{k-1}r_0\ ]$.

2. The GMRES method implementation.

At the $k^{\underline{th}}$ iteration of the GMRES method, k = 1, 2, ..., one seeks $z \in K_k$ for which $\| A(x_0 + z) - b \| = \| Az - r_0 \|$ is minimal. The $k^{\underline{th}}$ approximate solution is $x_k = x_0 + z$. It is shown below how to determine z using Householder transformations. We give first an overview and then a formal outline of the algorithm.

Finding $z \in K_k$ for which $\| Az - r_0 \|$ is minimal is equivalent to finding $y \in R^k$ for which $\| AK_k y - r_0 \|$ is minimal. Now $\| AK_k y - r_0 \| = \| K_{k+1} \begin{pmatrix} -1 \\ y \end{pmatrix} \|$; and so if $K_{k+1} = Q_{k+1} R_{k+1}$, then the problem is to minimize $\| Q_{k+1} R_{k+1} \begin{pmatrix} -1 \\ y \end{pmatrix} \| = \| R_{k+1} \begin{pmatrix} -1 \\ y \end{pmatrix} \| = \| -\beta e_1 + H_k y \|$, where $| \beta | = \| r_0 \|$ and H_k is an nxk upper Hessenberg matrix. If we write $H_k = \bar{Q}_k \bar{R}_k$, then the object is to minimize $\| -\bar{Q}_k^T (\beta e_1) + \bar{R}_k y \|$. Since $\bar{R}_k$ is n x k upper-triangular, the desired y is found by solving a k x k upper-triangular system with the first k components of $\bar{Q}_k^T (\beta e_1)$ as the right-hand side. It is not difficult to verify that if A is non-singular and $x_{k-1} \neq A^{-1}b$, then this upper-triangular system is non-singular. For the minimizing y, the norm of the residual is the norm of the last (n - k) components of $\bar{Q}_k^T (\beta e_1)$.

Here, we use Householder transformations to obtain the factorization $K_{k+1} = Q_{k+1} R_{k+1}$, i.e., Q_{k+1} is given implicitly by $Q_{k+1} = P_1 \ldots P_{k+1}$, where for i = 1, ..., k+1, $P_i = I - 2v_i v_i^T$,

$\| v_i \| = 1$. These transformations are chosen so that each P_i affects only components i through n of vectors on which it operates. We use Givens rotations to obtain the factorization $H_k = \bar{Q}_k \bar{R}_k$. That is, $\bar{Q}_k$ is given implicitly by $\bar{Q}_k = J_1^T \ldots J_k^T$. Each J_i is chosen to affect only the $i^{\underline{th}}$ and $(i+1)^{\underline{st}}$ components of vectors on which it operates. With $\bar{Q}_k$ and Q_{k+1} given in this way, the GMRES iterate $x_o + K_k y$ is determined from y as follows: First, note that $K_k = Q_k R_k$, where $Q_k = P_1 \ldots P_k$ and the columns of R_k are the first k columns of R_{k+1}. Second, note that $R_k = [\, \beta e_1, J_1^T \ldots J_{k-1}^T \bar{R}_{k-1} \,]$, where the columns of $\bar{R}_{k-1}$ are the first $(k-1)$ columns of $\bar{R}_k$. Then one has

$$x_o + K_k y = x_o + P_1 \ldots P_k \, [\, \beta e_1, J_1^T \ldots J_{k-1}^T \bar{R}_{k-1} \,] \, y.$$

In addition, note that $\bar{Q}_k^T (\beta e_1) = J_k \ldots J_1 (\beta e_1)$ has zero components below the $(k+1)^{\underline{st}}$, and so the residual norm is just $|J_k \ldots J_1 (\beta e_1)^{(k+1)}|$.

GMRES Algorithm:

1. Start: Suppose that x_o and a tolerance TOL are given.

   (a) Compute $K_1 = [r_o] = [b - Ax_o]$.

   (b) Determine P_1 such that $P_1 K_1 = [\beta e_1]$.

2. Iterate: For $k = 1, 2, \ldots, k_{max}$:

   (a) Suppose that one has

      (i) $P_1, \ldots, P_k$ such that

$$P_k \cdots P_1 K_k = \begin{cases} [\beta e_1], & \text{if } k = 1 \\ [\beta e_1, H_{k-1}], & \text{if } k > 1 ; \end{cases}$$

(ii) if $k > 1$, then $J_1, \ldots, J_{k-1}$ and $\bar{R}_{k-1}$ such that

$$J_{k-1} \cdots J_1 H_{k-1} = \bar{R}_{k-1} ;$$

(iii) $u = A^{k-1} r_o$;

(iv) $v = \begin{cases} \beta e_1, & \text{if } k = 1 \\ J_{k-1} \cdots J_1 (\beta e_1), & \text{if } k > 1 . \end{cases}$

(b) Overwrite $u \leftarrow Au$, and set $w = P_k \cdots P_1 u$.

(c) Determine P_{k+1} acting on components $k+1, \ldots, n$ such that $P_{k+1} w$ has zero components after the $(k+1)\underline{st}$.

(d) Overwrite $w \leftarrow P_{k+1} w$.

(e) If $k > 1$, overwrite $w \leftarrow J_{k-1} \cdots J_1 w$.

(f) Determine J_k acting on components k and $k+1$ such that $J_k w$ has zero components after the $k\underline{th}$.

(g) Overwrite $w \leftarrow J_k w$ and set $\bar{R}_k = [\bar{R}_{k-1}, w]$.

(h) Overwrite $v \leftarrow J_k v$.

(i) Set $r = |v^{(k+1)}|$; if $r \leq TOL$, go to (3).

3. Solve :

(a) Suppose that one has from (2)

(i) $P_1, \ldots, P_k$ such that

$$P_k \ldots P_1 K_K = \begin{cases} [\beta e_1], \text{ if } k = 1 \\ [\beta e_1, H_{k-1}], \text{ if } k > 1 ; \end{cases}$$

(ii) $J_1, \ldots, J_k$ and $\bar{R}_k$ such that

$$J_k \ldots J_1 H_k = \bar{R}_k ;$$

(iii) $v = J_k \ldots J_1 (\beta e_1)$;

(iv) the residual norm r.

(b) Determine y so that $\| v - \bar{R}_k y \|$ is minimal by solving a k x k triangular system with the first k rows of $\bar{R}_k$ as coefficient matrix and the first k components of v as right-hand side.

(c) Set

$$w = \begin{cases} P_1 [\beta e_1] y, \text{ if } k = 1 \\ P_1 \ldots P_k [\beta e_1, J_1^T \ldots J_{k-1}^T \bar{R}_{k-1}] y, \text{ if } k > 1, \end{cases}$$

where, when $k > 1$, $\bar{R}_{k-1}$ consists of the first (k - 1) columns of $\bar{R}_k$.

(d) Overwrite $x_0 \leftarrow x_0 + w$.

(e) If $r \leq TOL$, accept x_0 as the solution; otherwise, return to (1).

3. The Arnoldi Method Implementation.

At the $k^{\underline{th}}$ iteration of the Arnoldi method, $k = 1, 2, \ldots$, one tries to find $z \in K_k$ for which $[\, A(x_o + z) - b \,] = [\, Az - r_o \,]$ is orthogonal to K_k, i.e., for which $K_k^T [\, Az - r_o \,] = 0$. The k^{th} approximate solution is then $x_k = x_o + z$. We show how to determine such a z using Householder transformations.

Finding $z \in K_k$ for which $K_k^T [\, Az - r_o \,] = 0$ is equivalent to finding $y \in R^k$ for which

$$0 = K_k^T [\, AK_k y - r_o \,] = K_k^T K_{k+1} \begin{pmatrix} -1 \\ y \end{pmatrix} .$$

Let $P_1, \ldots, P_{k+1}$ be Householder transformations such that $P_{k+1} \ldots P_1 K_{k+1} = R_{k+1}$ and such that each P_i affects only components i through n of vectors on which it operates. Then $P_k \ldots P_1 K_k = R_k$, and it follows that the desired y satisfies

$$\begin{aligned} 0 &= R_k^T P_k \ldots P_1 \; P_1 \ldots P_{k+1} \; R_{k+1} \begin{pmatrix} -1 \\ y \end{pmatrix} \\ &= R_k^T P_{k+1} \; R_{k+1} \begin{pmatrix} -1 \\ y \end{pmatrix} . \end{aligned}$$

Since R_k has only zero rows after the $k^{\underline{th}}$, this becomes

$$0 = R_k^T \; R_{k+1} \begin{pmatrix} -1 \\ y \end{pmatrix} . \qquad (3.1)$$

One can show that if $r_{k-1} = b - A x_{k-1} \neq 0$, then R_k is of full rank. Thus if $r_{k-1} \neq 0$, then (3.1) holds if and only if the first k components of $R_{k+1} \begin{pmatrix} -1 \\ y \end{pmatrix}$ are zero. In this case, since $R_{k+1} \begin{pmatrix} -1 \\ y \end{pmatrix} = P_{k+1} \ldots P_1 K_{k+1} \begin{pmatrix} -1 \\ y \end{pmatrix}$ $= P_{k+1} \ldots P_1 [\, AK_k y - r_o \,]$, the absolute value of the $(k+1)^{st}$ component

of R_{k+1} $\binom{-1}{y}$ is the norm of the residual.

To determine y, we write $R_{k+1} = [\beta e_1, H_k]$, where $|\beta| = \| r_o \|$ and H_k is an n x k upper Hessenberg matrix. We assume that the first k rows of H_k constitute a nonsingular k x k matrix; otherwise, the method breaks down. If k = 1, then writing $H_1 = [h_1]$, one has $y = [\beta / h_1^{(1)}]$ with residual norm $\left|\beta h_1^{(2)} / h_1^{(1)}\right|$. If k > 1, then we choose Givens rotations $J_1, \ldots, J_{k-1}$ such that

$$J_{k-1} \cdots J_1 H_k = [\bar{R}_{k-1}, h_k],$$

where $\bar{R}_{k-1}$ is n x (k-1) upper triangular, and such that each J_i affects only components i and (i+1) of vectors on which it operates. Note that h_k has zero components after the $(k+1)^{\underline{st}}$. Set $v = J_{k-1} \cdots J_1 (\beta e_1)$. Then y is the solution of an upper-triangular system with the first k rows of $[\bar{R}_{k-1}, h_k]$ as coefficient matrix and with the first k components of v as right-hand side. The residual norm is $|(v^{(k)} h_k^{(k+1)}) / h_k^{(k)}|$.

Once y is known, the $k^{\underline{th}}$ Arnoldi iterate $x_o + K_k y$ is given by

$$x_o + K_k y = \begin{cases} x_o + P_1 [\beta e_1] y, \text{ if } k = 1 \\ x_o + P_1 \cdots P_k [\beta e_1, J_1^T \cdots J_{k-1}^T \bar{R}_{k-1}] y, \text{ if } k > 1. \end{cases}$$

Arnoldi Algorithm :

1. Start : Suppose that x_0 and a tolerance TOL are given.

   (a) Compute $K_1 = [r_o] = [b - Ax_o]$.

   (b) Determine P_1 such that $P_1 K_1 = [\beta e_1]$.

2. Iterate: For $k = 1, 2, \ldots, k_{max}$:

(a) Suppose that one has

(i) $P_1, \ldots, P_k$ such that

$$P_k \cdots P_1 K_k = \begin{cases} [\beta e_1], & \text{if } k = 1 \\ [\beta e_1, H_{k-1}], & \text{if } k > 1 ; \end{cases}$$

(ii) if $k = 2$, then h_1 such that $H_1 = [h_1]$;

if $k > 2$, then h_{k-1} with zero components after the $k^{\underline{th}}$ and $J_1, \ldots, J_{k-2}$ and $\bar{R}_{k-2}$ such that

$$J_{k-2} \cdots J_1 H_{k-1} = [\bar{R}_{k-2}, h_{k-1}]$$

(iii) $u = A^{k-1} r_o$;

(iv) $$v = \begin{cases} \beta e_1, & \text{if } k \le 2 \\ J_{k-2} \cdots J_1 (\beta e_1), & \text{if } k > 2 . \end{cases}$$

(b) Overwrite $u \leftarrow Au$, and set $w = P_k \cdots P_1 u$.

(c) Determine P_{k+1} acting on components $k+1, \ldots, n$ such that $P_{k+1} w$ has zero components after the $(k+1)^{\underline{st}}$.

(d) Overwrite $w \leftarrow P_{k+1} w$; if $k = 1$, set $h_1 = w$.

(e) If $k > 2$, overwrite $w \leftarrow J_{k-2} \cdots J_1 w$.

(f) If $k > 1$, determine J_{k-1} acting on components $(k-1)$ and k such that $J_{k-1} h_{k-1}$ has zero components after the $(k-1)^{\underline{st}}$.

(g) If $k > 1$, overwrite $h_{k-1} \leftarrow J_{k-1} h_{k-1}$,

set $h_k = J_{k-1} w$, and set

$$\bar{R}_{k-1} = \begin{cases} [h_{k-1}], \text{ if } k = 2 \\ [\bar{R}_{k-2}, h_{k-1}], \text{ if } k > 2 . \end{cases}$$

(h) If $k > 1$, overwrite $v \leftarrow J_{k-1} v$.

(i) Set $r = |(v^{(k)} h_k^{(k+1)}) / h_k^{(k)}|$; if $r \leq TOL$, go to (3) .

(3) Solve:

(a) Suppose that one has from (2)

(i) $P_1, \ldots, P_k$ such that

$$P_k \cdots P_1 K_k = \begin{cases} [\beta e_1], \text{ if } k = 1 \\ [\beta e_1, H_{k-1}], \text{ if } k > 1 ; \end{cases}$$

(ii) h_k with zero components after the $(k+1)^{\underline{st}}$, with $H_1 = [h_1]$, and, if $k > 1$, then $J_1, \ldots, J_{k-1}$ and $\bar{R}_{k-1}$ such that

$$J_{k-1} \cdots J_1 H_k = [\bar{R}_{k-1}, h_k];$$

(iii) $$v = \begin{cases} \beta e_1, \text{ if } k = 1 \\ J_{k-1} \cdots J_1 (\beta e_1), \text{ if } k > 1 ; \end{cases}$$

(iv) the residual norm r.

(b) Determine y by solving a k x k triangular system, the coefficient matrix of which consists of the first k rows of

$$\begin{cases} [h_1], \text{ if } k = 1 \\ [\bar{R}_{k-1}, h_k], \text{ if } k > 1 \end{cases}$$

and the right-hand side of which consists of the first k components of v.

(c) Set

$$w = \begin{cases} P_1 [\beta e_1] y, & \text{if } k = 1 \\ P_1 \cdots P_k [\beta e_1, J_1^T \cdots J_{k-1}^T \bar{R}_{k-1}] y, & \text{if } k > 1 . \end{cases}$$

(d) Overwrite $x_o \leftarrow x_o + w$.

(e) If $r \leq TOL$, accept x_o as the solution; otherwise, return to (1).

REFERENCES

1. A. Bjorck, "Solving linear least squares problems by Gram-Schmidt orthogonalization," BIT, 7 (1967), pp. 1-21.

2. P. N. Brown and A. C. Hindmarsh, "Matrix-free methods for stiff systems of ODE's," Lawrence Livermore National Laboratory Report UCRL-90770, Rev. 1, Feb. 1985; to appear in SIAM J. Num. Anal.

3. G. H. Golub and C. F. Van Loan, Matrix Computations, Johns Hopkins University Press, Baltimore, 1983.

4. Y. Saad, "Krylov subspace methods for solving large unsymmetric linear systems," Math. Comp., 37 (1981), pp. 105-126.

5. Y. Saad, "Practical use of some Krylov subspace methods for solving indefinite and nonsymmetric linear systems," SIAM J. Sci. Stat. Comput., 5 (1984), pp. 203-228.

6. Y. Saad and M. H. Schultz, "GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems," Yale University Computer Science Dept. Research Report YALEU/DCS/RR-254, Aug., 1983.

7. L. B. Wigton, N. J. Yu, and D. P. Young, "GMRES acceleration of fluid dynamics codes," Proceedings of the American Institute of Aeronautics and Astronautics 7th Computational Fluid Dynamics Conference, Cincinnati, Ohio, July 15-17, 1985.